

A Theoretical Analysis of Two-Stage Recommendation for Cold-Start Collaborative Filtering

Xiaoxue Zhao, Jun Wang

University College London, London, United Kingdom
`{x.zhao,j.wang}@cs.ucl.ac.uk`

Abstract. In this paper, we present a theoretical framework for tackling the cold-start collaborative filtering problem, where unknown targets (items or users) keep coming to the system, and there is a limited number of resources (users or items) that can be allocated and related to them. The solution requires a trade-off between exploitation and exploration as with the limited recommendation opportunities, we need to, on one hand, allocate the most relevant resources right away, but, on the other hand, it is also necessary to allocate resources that are useful for learning the target’s properties in order to recommend more relevant ones in the future. In this paper, we study a simple two-stage recommendation combining a sequential and a batch solution together. We first model the problem with the partially observable Markov decision process (POMDP) and provide an exact solution. Then, through an in-depth analysis over the POMDP value iteration solution, we identify that an exact solution can be abstracted as selecting resources that are not only highly relevant to the target according to the initial-stage information, but also highly correlated, either positively or negatively, with other *potential* resources for the next stage. With this finding, we propose an approximate solution to ease the intractability of the exact solution. Our initial results on synthetic data and the Movie Lens 100K dataset confirm the performance gains of our theoretical development and analysis.

1 Introduction

For approximately the last two decades, information retrieval has fundamentally transformed the way in which people seek and work with information. Roughly speaking, there are two types of information retrieval (IR) systems [5]. On one hand, we have *ad hoc* information retrieval, e.g., web search [24], which deals with a relatively fixed collection of information items (webpages, documents, images, product descriptions etc.) and dynamically changing users information requests. On the other hand, there are information filtering systems, such as content recommender systems, to address the situation where user profiles (as information requests) stay relatively static while new information items keep arriving. Nevertheless, in either case the fundamental problem remains the same, which is how to compute and find the *match* between the information items and information requests [23].

A more difficult scenario exists when there is little or no information about the request. For instance, in collaborative filtering (CF), it is hard to initialise recommendations when no past ratings are available. Research has been focused on the user *cold-start* problem [13,35], such as adopting a questionnaire stage [26,27,48], or an interactive procedure [47,13]. For the item cold-start problem, the main focus has

been put on utilising content information [30,14], which lies outside of the scope of CF, or experimental design [2].

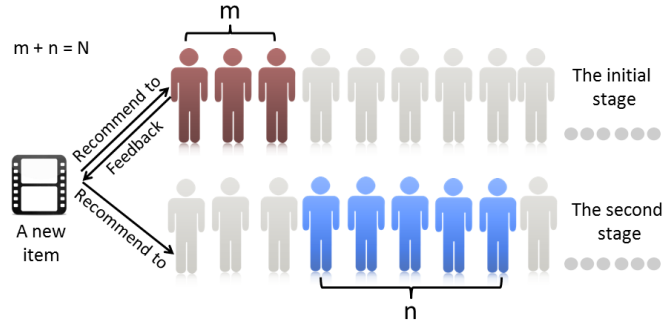
In our view, the cold-start problem can be regarded as a resource allocation problem, because in a short period of time, the number of recommendations (for a new item or to a new user) is usually much smaller than the size of the available pool. Thus only a small portion can be selected due to the limited resources. For example, advertisements of a new item can only be sent to a limited number of users, whereas a new user can only rate a limited number of items after joining a web service. Therefore, it is important to utilise the limited recommendation resources wisely.

In this paper, we formulate and analyse a simple yet practical two-stage process to solve the recommendation allocation problem. During the initial stage, we use a portion of recommendation allocations to estimate the new item's (user's) model. After that, during the second stage, we recommend the item (user) using the remaining resources. We argue that the goal of this process should be to maximise the *total* feedback over two stages, which leads to a trade-off between exploitation and exploration. This means that, with limited resources, we should not separate the learning process from recommendation. Rather, recommendations should be made right from the beginning while also intelligently accommodating the learning requirement. The proposed two-stage recommendation process is depicted in Figure 1. In CF, items and users are usually modeled symmetrically [44,22], and, as such, we will focus on the item cold-start problem as our working example. However, all the analysis can be easily adapted to a user cold-start scenario.

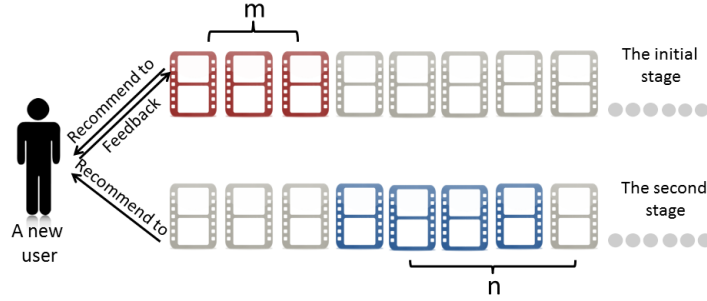
Dividing the recommendation process into two stages is simple yet powerful as it combines both batch and feedback mechanisms together. The motivations for our analysis on the setting are threefold. First, for the cold-start item, to learn its profile over time, one way is to sequentially target the item to one user, observe its feedback, update the item profile model, and find another user with the updated model, in an interactive manner similar to [47]. However, as users differ in their response times, waiting one user's response before targeting to the other is practically infeasible in many cases. Second, it may also be computationally too expensive for the system to update whenever a new rating is registered. A two-stage process, by contrast, enables the system to act economically. Third, statistically analysing the separated two-stage process also enables a clear understanding of the trade-off between exploitation-exploration (EE) embedded in many practical applications.

The two-stage setup also covers a variety of other applications. For example, in IR, when a query is issued, the system shows two subsequent pages to the user such that the second-page results can be refined [36,19]. And, in online display advertising, for a new campaign, in order to understand which part of users should be targeted, the advertiser can spend some budget to show the ads to different users and collect their feedback (i.e., ad click or conversion), and then after the warming-up stage, leverage the users' feedback and refine the target user groups for higher advertising performance [46].

We first formulate the two-stage recommendation with a POMDP framework. We then derive the exact solution for both a correlated-user model (CU) and a matrix factorisation (MF) model, along with a discussion on the link between them. After that, we present our theoretical finding, i.e., the users to choose in the initial stage should be those not only highly relevant according to the initial-stage information, but also able to potentially guide us to find users with high expected values in the next



(a) For a cold-start item.



(b) For a cold-start user.

Fig. 1: Schematic figures of the two-stage recommendation process for (a) a cold-start item, and (b) a cold-start user. The total N resources are allocated in two stages. In the initial stage, m users (items) are selected, with their feedback used to update the profile of the new item (user). Then another n users (items) are selected in the second stage to exploit the updated profile. The target is to maximise the overall feedback over two stages.

stage with updated information. This ability of guidance can be further abstracted as a strong correlation between the initial-stage users and potential second-stage users, no matter positive or negative. With this finding, we propose the approximation method *guided* exploitation-exploration (GEE). We argue that, as our objective differs from that of an upper confidence bound (UCB) or an active learning (AL) approach, our proposed GEE algorithm is significantly different from them. The effectiveness of the proposed solution is confirmed by our experiments, conducted using both synthetic data and a real dataset.

The rest of this paper is organised as follows. We formulate the problem and present its exact solution in Section 2. In Section 3, we present the proposed approximate solution GEE and afterwards we discuss the related previous work in Section 4. Our experimental results are reported in Section 5, and Section 6 summarises and concludes this paper.

2 The Two-Stage Model

In this section, we formulate CF into the POMDP framework, which will lead us to the exact solution of our problem. A POMDP models a Markov decision process where the true current state of the system is partially unobservable [20]. In the scenario of the item cold-start recommendation, the true state is each user's genuine

Table 1: Summary of key notations.

Notation	Description
\mathcal{U}	The entire user set
$t \in \{1, 2\}$	The stage (timestep) of the process
m, n	The number of users to select at the initial stage and the second stage respectively
\mathbf{u}, \mathbf{v}	The users to choose in the initial stage and second stage respectively
$\backslash \mathbf{u}$	The users not selected in the initial stage, $\backslash \mathbf{u} = \mathcal{U} \backslash \mathbf{u}$
\mathbf{R}	The preferences (a random vector) of all users over the item under consideration
$\mathbf{R}_{\mathbf{u}}, \mathbf{R}_{\mathbf{v}}$	\mathbf{R} partitioned by \mathbf{u} and \mathbf{v} respectively
$\mathbf{r}_{\mathbf{u}}, \mathbf{r}_{\mathbf{v}}$	Feedback from \mathbf{u} and \mathbf{v} respectively
$\boldsymbol{\theta}^{(t)}, \boldsymbol{\Phi}^{(t)}, \mathbf{C}^{(t)}$	The mean, covariance matrix, correlation matrix of \mathbf{R} at time t (CU model)
$\rho_{i,j}^{(t)}$	Correlation between u and v at t
\mathbf{P}	The matrix with each row as a user vector (MF model)
\mathbf{q}	The target item's feature vector (MF model)
$\boldsymbol{\nu}^{(t)}, \boldsymbol{\Psi}^{(t)}$	The mean and covariance matrix of the item vector at time t (MF model)
T	The sampling number

(potential) preference as to the new item, which is unknown for the users having not rated it. To model the decision process, we start with a correlated-user (CU) model as a probabilistic description of the memory-based models in CF [17,10] and formulate it with POMDP. Then, we decompose the user-item rating matrix to gain its formulation in the domain of MF. We provide, for each model, the exact solution on how to select users optimally in order to collect maximal overall feedback from the users over two stages.

2.1 Correlated-User Model with POMDP

The CU model with POMDP (CU-POMDP) is depicted in Figure 2. Let us denote the available user pool as \mathcal{U} . For each new item that joins the system, the recommendation system should make the following decisions: in the initial stage, choose an initial m users to start with, collect their feedback, and update the system's belief state; and in the second stage, choose another n users to exploit the information gained from the initial stage. $N = m + n$ is the total number of users that the item is to be targeted to. For the reader's convenience, we provide a list of key notations used in this paper in Table 1. We consider only one cold-start item, but the scenario is similar if multiple cold-start items are present.

Our goal is to find the optimal policy that can maximise the expected total ratings over two stages. To capture the relations between users' preferences, we model the preferences of all users, denoted by \mathbf{R} , to follow a multivariate Gaussian distribution

$$p^{(t)}(\mathbf{R}) \sim \mathcal{N}(\boldsymbol{\theta}^{(t)}, \boldsymbol{\Phi}^{(t)}), \quad t \in \{1, 2\} \quad (1)$$

with its mean and covariance matrix as $\boldsymbol{\theta}^{(t)}$ and $\boldsymbol{\Phi}^{(t)}$. The distribution above is the system's belief over the true state \mathbf{R} at each stage t , referred to as the belief state according to POMDP. By recommending the item to users and receiving their feedback, the belief state evolves from $p^{(1)}(\mathbf{R})$ to $p^{(2)}(\mathbf{R})$. Our problem is a POMDP

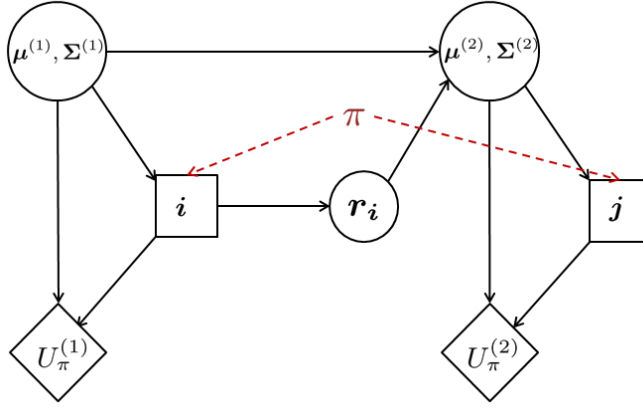


Fig. 2: The two-stage CU-POMDP as illustrated by an influence diagram, with respect to the correlated-user model. Circular nodes are random variables and square nodes are the recommendation decision, and the rhombus nodes are the utility at each stage.

because the true preferences \mathbf{R} are unknown (or only partially known), but can be modelled through a distribution.

This model is non-trivial because it has utilised all user-user correlations via a multi-variate Gaussian model. To obtain the belief state for the initial stage, we can impose an i.i.d. assumption on the users' preferences on different items. As such, $\theta^{(1)}$ can be estimated by the users' mean ratings, and $\Phi^{(1)}$ can be estimated by the user-user covariances on previously co-rated items. To emphasise the role of user-user correlation, in the following, we also make use of the following representation

$$\begin{aligned}\Phi^{(1)} &= \text{Dg}[\Phi^{(1)}]^{1/2} \mathbf{C}^{(1)} \text{Dg}[\Phi^{(1)}]^{1/2} \\ &= \text{diag}[\phi^{(1)}] \mathbf{C}^{(1)} \text{diag}[\phi^{(1)}]\end{aligned}\quad (2)$$

where $\text{Dg}(\Phi^{(1)})$ denotes the diagonal matrix with the same diagonal elements of $\Phi^{(1)}$, $\phi^{(1)}$ denotes the vector formed by the users' standard deviations of ratings ($\phi^{(1)} = \text{diag}[\text{Dg}^{1/2}(\Phi^{(1)})]$), and $\mathbf{C}^{(1)}$ is the correlation matrix whose element $\rho_{u,v}^{(1)}$ is the correlation between user u and user v .

A policy π is defined to make the decision at each stage on the basis of the available information:

$$\mathbf{u} = \pi(\theta^{(1)}, \Phi^{(1)}, \mathcal{U}), \text{ and} \quad (3)$$

$$\mathbf{v} = \pi(\theta^{(2)}, \Phi^{(2)}, \mathcal{U} \setminus \mathbf{u}), \quad (4)$$

where we use vectors \mathbf{u} and \mathbf{v} to denote the user selection decisions for the two stages respectively ($|\mathbf{u}| = m$ and $|\mathbf{v}| = n$). Here we also use the constraint that the target item should not be recommended repeatedly to the same user. Therefore, the available user pool will be the remaining users $\mathcal{U} \setminus \mathbf{u}$ for the second stage. The total expected ratings collected at each stage is the element-wise summation of the expected rating vector of each selection, which we refer to as reward $U_\pi^{(t)}$

$$U_\pi^{(1)} = \mathbb{E}^{(1)}[\mathbf{1}^T \mathbf{R}_\mathbf{u}], \quad (5)$$

$$U_\pi^{(2)} = \mathbb{E}^{(2)}[\mathbf{1}^T \mathbf{R}_\mathbf{v}]. \quad (6)$$

We use \mathbf{R}_u (\mathbf{R}_v) to denote the random vector \mathbf{R} partitioned by user selections \mathbf{u} (\mathbf{v}). We will use the same partition rule throughout this paper.

The objective is to find a policy of selecting users such that the expected *total* reward of the two stages are maximised

$$\pi^* = \arg \max_{\pi} (U_{\pi}^{(1)} + U_{\pi}^{(2)}). \quad (7)$$

2.1.1 Belief Update Let us consider the problem in a reverse order. Suppose the system has already recommended the item to m users in the initial stage and received feedback \mathbf{r}_u . Given the feedback, the system can update its belief state on the remaining users $\mathcal{U} \setminus \mathbf{u}$ (simplified as $\setminus \mathbf{u}$) by the conditional multivariate Gaussian distribution, conditioned on the observations

$$p^{(2)}(\mathbf{R}_{\setminus \mathbf{u}}) \sim \mathcal{N}(\boldsymbol{\theta}_{\setminus \mathbf{u}}^{(2)}, \boldsymbol{\Phi}_{\setminus \mathbf{u}, \setminus \mathbf{u}}^{(2)}), \text{ where} \quad (8)$$

$$\boldsymbol{\theta}_{\setminus \mathbf{u}}^{(2)} = \boldsymbol{\theta}_{\setminus \mathbf{u}}^{(1)} + \boldsymbol{\Phi}_{\setminus \mathbf{u}, \mathbf{u}}^{(1)} [\boldsymbol{\Phi}_{\mathbf{u}, \mathbf{u}}^{(1)}]^{-1} (\mathbf{r}_u - \boldsymbol{\theta}_{\mathbf{u}}^{(1)}) \quad (9)$$

$$\boldsymbol{\Phi}_{\setminus \mathbf{u}, \setminus \mathbf{u}}^{(2)} = \boldsymbol{\Phi}_{\setminus \mathbf{u}, \setminus \mathbf{u}}^{(1)} - \boldsymbol{\Phi}_{\setminus \mathbf{u}, \mathbf{u}}^{(1)} [\boldsymbol{\Phi}_{\mathbf{u}, \mathbf{u}}^{(1)}]^{-1} \boldsymbol{\Phi}_{\mathbf{u}, \setminus \mathbf{u}}^{(1)}. \quad (10)$$

To gain insight with the view of correlated users, we reformulate the update functions with the correlation matrix $\mathbf{C}^{(1)}$ as follows. According to Eq. (2), we obtain

$$[\boldsymbol{\Phi}_{\mathbf{u}, \mathbf{u}}^{(1)}]^{-1} = \text{diag}[\boldsymbol{\phi}_{\mathbf{u}}^{(1)}]^{-1} [\mathbf{C}_{\mathbf{u}, \mathbf{u}}^{(1)}]^{-1} \text{diag}[\boldsymbol{\phi}_{\mathbf{u}}^{(1)}]^{-1}, \text{ and} \quad (11)$$

$$[\boldsymbol{\Phi}_{\setminus \mathbf{u}, \mathbf{u}}^{(1)}] = \text{diag}[\boldsymbol{\phi}_{\setminus \mathbf{u}}^{(1)}] \mathbf{C}_{\setminus \mathbf{u}, \mathbf{u}}^{(1)} \text{diag}[\boldsymbol{\phi}_{\mathbf{u}}^{(1)}]. \quad (12)$$

Substituting Eqs. (12) and (11) into (9) we further get

$$\boldsymbol{\theta}_{\setminus \mathbf{u}}^{(2)} = \boldsymbol{\theta}_{\setminus \mathbf{u}}^{(1)} + \text{diag}[\boldsymbol{\phi}_{\setminus \mathbf{u}}^{(1)}] \mathbf{C}_{\setminus \mathbf{u}, \mathbf{u}}^{(1)} [\mathbf{C}_{\mathbf{u}, \mathbf{u}}^{(1)}]^{-1} \text{diag}[\boldsymbol{\phi}_{\mathbf{u}}^{(1)}]^{-1} (\mathbf{r}_u - \boldsymbol{\theta}_{\mathbf{u}}^{(1)}) \quad (13)$$

Particularly, if we assume equal rating variance for all users, and disregard the correlations among \mathbf{u} such that $\mathbf{C}_{\mathbf{u}, \mathbf{u}}^{(1)}$ becomes an identity matrix, then Eq. (13) reduces to a weighted summation of the observed ratings centred by their prior expectations $\mathbf{r}_u - \boldsymbol{\theta}_{\mathbf{u}}^{(1)}$, with the weights as the correlations between unobserved users and observed users

$$\boldsymbol{\theta}_{\setminus \mathbf{u}}^{(2)} = \boldsymbol{\theta}_{\setminus \mathbf{u}}^{(1)} + \mathbf{C}_{\setminus \mathbf{u}, \mathbf{u}}^{(1)} (\mathbf{r}_u - \boldsymbol{\theta}_{\mathbf{u}}^{(1)}). \quad (14)$$

Eq. (14) looks very familiar to us because it simulates the popular memory-based (user-based) CF algorithm, which takes the neighbours' ratings regarding the target item, centres them by the mean ratings of the neighbours, and estimates the target user's preference regarding this item as their weighted summation [17], where Pearson correlation is commonly used to calculate the weights [30]. We thus see the user-based recommendation heuristic as an approximation of our CU model.

From the above formula we can see that: (i) by observing users \mathbf{u} in the initial stage, the expectations of unobserved users are also updated; (ii) the covariances (correlations) between observed and unobserved users act as the bridge through which feedback from selected users can update our belief regarding other users.

2.1.2 Exact Solution To obtain the exact solution, consider $V^*(\boldsymbol{\theta}^{(t)}, \boldsymbol{\Phi}^{(t)}, \mathcal{T})$ which is the maximally achievable expected total future reward with current information $\boldsymbol{\theta}^{(t)}$, $\boldsymbol{\Phi}^{(t)}$ and remaining steps ($\mathcal{T} = 1, 2$). With the updated belief according to Eq. (8) already *given*, the optimal expected reward for the second stage is simply a greedy approach:

$$\begin{aligned} V_{\text{CU}}^*(\boldsymbol{\theta}^{(2)}, \boldsymbol{\Phi}^{(2)}, 1) &= \max_{\pi} U_{\pi}^{(2)} \\ &= \max_{\mathbf{v} \subset \mathcal{U} \setminus \mathbf{u}} \mathbb{E}^{(2)}[\mathbf{1}^T \mathbf{R}_{\mathbf{v}}] \\ &= \max_{\mathbf{v} \subset \mathcal{U} \setminus \mathbf{u}} \mathbf{1}^T \boldsymbol{\theta}_{\mathbf{v}}^{(2)}. \end{aligned} \quad (15)$$

By working backwards the total maximal expected reward for two stages can be obtained as

$$\begin{aligned} V_{\text{CU}}^*(\boldsymbol{\theta}^{(1)}, \boldsymbol{\Phi}^{(1)}, 2) &= \max_{\pi} (U_{\pi}^{(1)} + U_{\pi}^{(2)}) \\ &= \max_{\mathbf{u} \subset \mathcal{U}} \left(\mathbb{E}^{(1)}[\mathbf{1}^T \mathbf{R}_{\mathbf{u}} + V_{\text{CU}}^*(\boldsymbol{\theta}^{(2)}, \boldsymbol{\Phi}^{(2)}, 1)] \right) \\ &= \max_{\mathbf{u} \subset \mathcal{U}} \left(\mathbb{E}^{(1)}[\mathbf{1}^T \mathbf{R}_{\mathbf{u}}] + \int p^{(1)}(\mathbf{R}_{\mathbf{u}} = \mathbf{r}_{\mathbf{u}}) V_{\text{CU}}^*(\boldsymbol{\theta}^{(2)}, \boldsymbol{\Phi}^{(2)}, 1) d\mathbf{r}_{\mathbf{u}} \right). \end{aligned} \quad (16)$$

Substituting Eqs. (15) and (9) into (16) we reach the exact solution obtained by value iteration:

$$\begin{aligned} V_{\text{CU}}^*(\boldsymbol{\theta}^{(1)}, \boldsymbol{\Phi}^{(1)}, 2) &= \max_{\mathbf{u} \subset \mathcal{U}} \left\{ \underbrace{\mathbf{1}^T \boldsymbol{\theta}_{\mathbf{u}}^{(1)}}_{\text{exploitation}} + \underbrace{\int p^{(1)}(\mathbf{R}_{\mathbf{u}} = \mathbf{r}_{\mathbf{u}}) \max_{\mathbf{v} \subset \mathcal{U} \setminus \mathbf{u}} \left[\mathbf{1}^T \left(\boldsymbol{\theta}_{\mathbf{v}}^{(1)} + \boldsymbol{\Phi}_{\mathbf{v}, \mathbf{u}}^{(1)} [\boldsymbol{\Phi}_{\mathbf{u}, \mathbf{u}}^{(1)}]^{-1} (\mathbf{r}_{\mathbf{u}} - \boldsymbol{\theta}_{\mathbf{u}}^{(1)}) \right) \right] d\mathbf{r}_{\mathbf{u}}}_{\text{exploration}} \right\}. \end{aligned} \quad (17)$$

Eq. (17) suggests that the merit of choosing users \mathbf{u} at the initial stage lies in two components:

- **Exploitation.** It is the immediate expected reward, denoted by $\mathbf{1}^T \boldsymbol{\theta}_{\mathbf{u}}^{(1)}$, determined by the prior information on the users.
- **Exploration.** The exploration component shows how the feedback from users \mathbf{u} can lead the system to find optimal selections with updated knowledge. Consider that the feedback deviates from the prior information such that $(\mathbf{r}_{\mathbf{u}} - \boldsymbol{\theta}_{\mathbf{u}}^{(1)}) \neq \mathbf{0}$, the updated belief state will then lead us to find users which bring “extra” returns via the term $\boldsymbol{\Phi}_{\mathbf{v}, \mathbf{u}}^{(1)} [\boldsymbol{\Phi}_{\mathbf{u}, \mathbf{u}}^{(1)}]^{-1} (\mathbf{r}_{\mathbf{u}} - \boldsymbol{\theta}_{\mathbf{u}}^{(1)})$. No matter the deviation is positive or negative, we can always benefit from it by selecting corresponding optimal users in the second stage. As mentions above, this term relates to correlations between the users of the two stages. The larger the correlations are, the more the system can *gain* from the discrepancy between the observations and the prior information.

2.2 Matrix Factorization Model with POMDP

To gain insights from the formulation of latent factor models, consider MF with POMDP (MF-POMDP). For this purpose, we use the probabilistic model $\mathbf{R} = \mathbf{P}\mathbf{q} + \xi$ such that $\mathbf{P} = (\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_{|\mathcal{U}|})^T$ is a $|\mathcal{U}| \times K$ matrix containing the users’ information, \mathbf{q} is a K -dimensional item vector, and ξ is a random variable with zero mean and

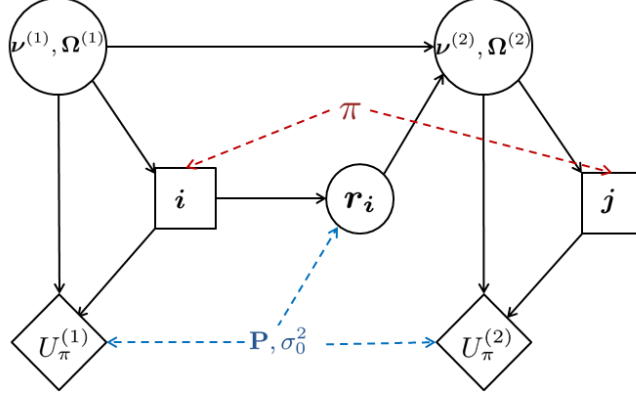


Fig. 3: The two-stage MF-POMDP as illustrated by an influence diagram, with respect to the matrix factorization model.

variance σ_0^2 . If we assume fixed user vectors \mathbf{P} and unknown item vector \mathbf{q} [47,37], CU-POMDP is translated to a decision process under the belief state of the unobservable item vector (see Figure 3)

$$p^{(t)}(\mathbf{q}) \sim \mathcal{N}(\boldsymbol{\nu}^{(t)}, \boldsymbol{\Psi}^{(t)}), \quad (18)$$

where $\boldsymbol{\nu}^{(1)}$ and $\boldsymbol{\Psi}^{(1)}$ are the mean and covariance matrix of the item vector. The belief state over the item vector then determines the belief over the preferences of users

$$p^{(t)}(\mathbf{R}) \sim \mathcal{N}(\mathbf{P}\boldsymbol{\nu}^{(t)}, \mathbf{P}\boldsymbol{\Psi}^{(t)}\mathbf{P}^T + \sigma_0^2\mathbf{I}). \quad (19)$$

By observing users \mathbf{u} with feedback \mathbf{r}_u the belief state can be updated according to the Bayes rule

$$p^{(2)}(\mathbf{q}) \sim \mathcal{N}(\boldsymbol{\nu}^{(2)}, \boldsymbol{\Psi}^{(2)}), \text{ where} \quad (20)$$

$$\boldsymbol{\nu}^{(2)} = \boldsymbol{\nu}^{(1)} + \boldsymbol{\Psi}^{(1)}\mathbf{P}_u^T(\mathbf{P}_u\boldsymbol{\Psi}^{(1)}\mathbf{P}_u^T + \sigma_0^2\mathbf{I})^{-1}(\mathbf{r}_u - \mathbf{P}_u\boldsymbol{\nu}^{(1)}), \quad (21)$$

$$\boldsymbol{\Psi}^{(2)} = [(\boldsymbol{\Psi}^{(1)})^{-1} + \mathbf{P}_u^T\mathbf{P}_u/\sigma_0^2]^{-1}. \quad (22)$$

Thus,

$$\begin{aligned} \mathbb{E}^{(2)}(\mathbf{R}_{\setminus u}|\mathbf{r}_u) &= \mathbf{P}_{\setminus u}\boldsymbol{\nu}^{(2)} \\ &= \mathbf{P}_{\setminus u}\boldsymbol{\nu}^{(1)} + \mathbf{P}_{\setminus u}\boldsymbol{\Psi}^{(1)}\mathbf{P}_u^T(\mathbf{P}_u\boldsymbol{\Psi}^{(1)}\mathbf{P}_u^T + \sigma_0^2\mathbf{I})^{-1}(\mathbf{r}_u - \mathbf{P}_u\boldsymbol{\nu}^{(1)}). \end{aligned} \quad (23)$$

Comparing Eq. (23) with Eq. (9) we find a nice alignment between the two models. Actually, by dimension reduction the covariance between user u 's and user v 's ratings can be translated as

$$\Phi_{u,v}^{(1)} = \mathbf{p}_u^T\boldsymbol{\Psi}^{(1)}\mathbf{p}_v, \quad (24)$$

when σ_0^2 is very small compared to the covariance between the two users's true preferences ($\sigma_0^2 \ll \mathbf{p}_u^T\boldsymbol{\Psi}^{(1)}\mathbf{p}_v$). Eq. (24) has converted the statistical property (the covariance of preferences between the two users) into the function of the feature vectors of the two users.

By the same token, we write the optimal value function for the MF-POMDP as

$$\begin{aligned}
V_{\text{MF}}^*(\boldsymbol{\nu}^{(1)}, \boldsymbol{\Psi}^{(1)}, 2) = \max_{\mathbf{u} \in \mathcal{U}} & \left\{ \mathbf{1}^T \mathbf{P}_{\mathbf{u}} \boldsymbol{\nu}^{(1)} + \right. \\
& \int p^{(1)}(\mathbf{R}_{\mathbf{u}} = \mathbf{r}_{\mathbf{u}}) \max_{\mathbf{v} \in \mathcal{U} \setminus \mathbf{u}} \left[\mathbf{1}^T \left(\mathbf{P}_{\mathbf{v}} \boldsymbol{\nu}^{(1)} + \right. \right. \\
& \left. \left. \mathbf{P}_{\mathbf{v}} \boldsymbol{\Psi}^{(1)} \mathbf{P}_{\mathbf{u}}^T [\mathbf{P}_{\mathbf{u}} \boldsymbol{\Psi}^{(1)} \mathbf{P}_{\mathbf{u}}^T + \sigma_0^2 \mathbf{I}]^{-1} (\mathbf{r}_{\mathbf{u}} - \mathbf{P}_{\mathbf{u}} \boldsymbol{\nu}^{(1)}) \right) d\mathbf{r}_{\mathbf{u}} \right] \left. \right\}. \tag{25}
\end{aligned}$$

2.3 A Toy Example

Let us look at a simple three-user case and its analytical solution. In this example, one user is selected in each stage. We base this example on the CU model so that the effect of user-user correlation can be illustrated more straightforwardly.

Suppose

$$\boldsymbol{\theta}^{(1)} = \begin{pmatrix} \theta_1^{(1)} \\ \theta_2^{(1)} \\ \theta_3^{(1)} \end{pmatrix}, \quad \boldsymbol{\Phi}^{(1)} = \begin{pmatrix} \Phi_{1,1}^{(1)} & \Phi_{1,2}^{(1)} & \Phi_{1,3}^{(1)} \\ \Phi_{2,1}^{(1)} & \Phi_{2,2}^{(1)} & \Phi_{2,3}^{(1)} \\ \Phi_{3,1}^{(1)} & \Phi_{3,2}^{(1)} & \Phi_{3,3}^{(1)} \end{pmatrix}.$$

Without loss of generality, we assume $\Phi_{1,3}^{(1)} > \Phi_{1,2}^{(1)} > \Phi_{2,3}^{(1)}$ (and ignore the case with equal covariance for now). Suppose user 1 is selected in the initial stage with the observation as r_1 , the update for the second and the third users are,

$$\begin{aligned}
\theta_2^{(2)}(r_1) &= \theta_2^{(1)} + \Phi_{2,1}^{(1)} (\Phi_{1,1}^{(1)})^{-1} (r_1 - \theta_1^{(1)}), \\
\theta_3^{(2)}(r_1) &= \theta_3^{(1)} + \Phi_{3,1}^{(1)} (\Phi_{1,1}^{(1)})^{-1} (r_1 - \theta_1^{(1)}).
\end{aligned}$$

By introducing $z_1 = (r_1 - \theta_1^{(1)}) / \sqrt{\Phi_{1,1}^{(1)}}$, the above updates become

$$\begin{aligned}
\theta_2^{(2)}(z_1) &= \theta_2^{(1)} + \Phi_{2,1}^{(1)} (\Phi_{1,1}^{(1)})^{-1/2} z_1, \\
\theta_3^{(2)}(z_1) &= \theta_3^{(1)} + \Phi_{3,1}^{(1)} (\Phi_{1,1}^{(1)})^{-1/2} z_1.
\end{aligned}$$

We can see that both $\theta_2^{(2)}$ and $\theta_3^{(2)}$ are linear in z_1 . The turning point between choosing user 2 and user 3 is obtained when the above two are equal to each other, which is at

$$d_1 = \frac{\theta_2^{(1)} - \theta_3^{(1)}}{\Phi_{3,1}^{(1)} - \Phi_{2,1}^{(1)}} \sqrt{\Phi_{1,1}^{(1)}}.$$

Since $\Phi_{3,1}^{(1)} > \Phi_{2,1}^{(1)}$, if $z_1 > d_1$, user 3 should be selected whereas if $z_1 < d_1$ user 2 should be selected in the second stage. Thus, the optimal reward when choosing user 1 at the initial stage is

$$\begin{aligned}
V_{u=1}^*(\theta^{(1)}, \Phi^{(1)}, 2) &= \\
&\theta_1^{(1)} + \int p^{(1)}(r_1) \cdot \max_{v=2,3} \left(\theta_v^{(1)} + \Phi_{v,1}^{(1)} (\Phi_{1,1}^{(1)})^{-1} (r_1 - \theta_1^{(1)}) \right) dr_1 \\
&= \theta_1^{(1)} + \int_{-\infty}^{d_1} p^{(1)}(z_1) \left[\theta_2^{(1)} + \Phi_{2,1}^{(1)} (\Phi_{1,1}^{(1)})^{-1/2} z_1 \right] dz_1 \\
&\quad + \int_{d_1}^{\infty} p^{(1)}(z_1) \left[\theta_3^{(1)} + \Phi_{3,1}^{(1)} (\Phi_{1,1}^{(1)})^{-1/2} z_1 \right] dz_1 \\
&= \theta_1^{(1)} + 1/2(\theta_2^{(1)} + \theta_3^{(1)}) + 1/2(\theta_2^{(1)} - \theta_3^{(1)}) \operatorname{erf}\left(\frac{d_1}{\sqrt{2}}\right) \\
&\quad - \frac{1}{\sqrt{2\pi}} \frac{\Phi_{2,1}^{(1)} - \Phi_{3,1}^{(1)}}{\sqrt{\Phi_{1,1}^{(1)}}} \mathbf{e}^{-\frac{d_1^2}{2}}.
\end{aligned}$$

Similarly,

$$\begin{aligned}
V_{u=2}^*(\theta^{(1)}, \Phi^{(1)}, 2) &= \\
&\theta_2^{(1)} + \int p^{(1)}(r_2) \cdot \max_{v=3,1} \left(\theta_v^{(1)} + \Phi_{v,2}^{(1)} (\Phi_{2,2}^{(1)})^{-1} (r_2 - \theta_2^{(1)}) \right) dr_2 \\
&= \theta_2^{(1)} + \int_{-\infty}^{d_2} p^{(1)}(z_2) \left[\theta_3^{(1)} + \Phi_{3,2}^{(1)} (\Phi_{2,2}^{(1)})^{-1/2} z_2 \right] dz_2 \\
&\quad + \int_{d_2}^{\infty} p^{(1)}(z_2) \left[\theta_1^{(1)} + \Phi_{1,2}^{(1)} (\Phi_{2,2}^{(1)})^{-1/2} z_2 \right] dz_2 \\
&= \theta_2^{(1)} + 1/2(\theta_3^{(1)} + \theta_1^{(1)}) + 1/2(\theta_3^{(1)} - \theta_1^{(1)}) \operatorname{erf}\left(\frac{d_2}{\sqrt{2}}\right) \\
&\quad - \frac{1}{\sqrt{2\pi}} \frac{\Phi_{3,2}^{(1)} - \Phi_{1,2}^{(1)}}{\sqrt{\Phi_{2,2}^{(1)}}} \mathbf{e}^{-\frac{d_2^2}{2}},
\end{aligned}$$

$$\begin{aligned}
V_{u=3}^*(\theta^{(1)}, \Phi^{(1)}, 2) &= \\
&\theta_3^{(1)} + \int p^{(1)}(r_3) \cdot \max_{v=1,2} \left(\theta_v^{(1)} + \Phi_{v,3}^{(1)} (\Phi_{3,3}^{(1)})^{-1} (r_3 - \theta_3^{(1)}) \right) dr_3 \\
&= \theta_3^{(1)} + \int_{-\infty}^{d_3} p^{(1)}(z_3) \left[\theta_2^{(1)} + \Phi_{2,3}^{(1)} (\Phi_{3,3}^{(1)})^{-1/2} z_3 \right] dz_3 \\
&\quad + \int_{d_3}^{\infty} p^{(1)}(z_3) \left[\theta_1^{(1)} + \Phi_{1,3}^{(1)} (\Phi_{3,3}^{(1)})^{-1/2} z_3 \right] dz_3 \\
&= \theta_3^{(1)} + 1/2(\theta_1^{(1)} + \theta_2^{(1)}) + 1/2(\theta_2^{(1)} - \theta_1^{(1)}) \operatorname{erf}\left(\frac{d_3}{\sqrt{2}}\right) \\
&\quad - \frac{1}{\sqrt{2\pi}} \frac{\Phi_{2,3}^{(1)} - \Phi_{1,3}^{(1)}}{\sqrt{\Phi_{3,3}^{(1)}}} \mathbf{e}^{-\frac{d_3^2}{2}},
\end{aligned}$$

where

$$d_2 = \frac{\theta_3^{(1)} - \theta_1^{(1)}}{\Phi_{1,2}^{(1)} - \Phi_{3,2}^{(1)}} \sqrt{\Phi_{2,2}^{(1)}}, \quad d_3 = \frac{\theta_2^{(1)} - \theta_1^{(1)}}{\Phi_{1,3}^{(1)} - \Phi_{2,3}^{(1)}} \sqrt{\Phi_{3,3}^{(1)}}.$$

Note that the above formula are not rotational symmetric due to the asymmetry caused by $\Phi_{1,3}^{(1)} > \Phi_{1,2}^{(1)} > \Phi_{2,3}^{(1)}$.

To illustrate the results, let us look at a numerical example according to the above solutions. Suppose

$$\theta^{(1)} = \begin{pmatrix} 3.2 \\ 2.5 \\ 3.5 \end{pmatrix}, \quad \Phi^{(1)} = \begin{pmatrix} 1.6 & 0.25 & 1.6 \\ 0.25 & 3.2 & 0.20 \\ 1.6 & 0.20 & 3.5 \end{pmatrix}.$$

The correlation matrix is thus

$$\mathbf{C}^{(1)} = \begin{pmatrix} 1 & 0.11 & 0.68 \\ 0.11 & 1 & 0.06 \\ 0.68 & 0.06 & 1 \end{pmatrix}.$$

When user 1 is selected at the initial stage:

$$\begin{aligned} \theta_2^{(2)}(r_1) &= \theta_2^{(1)} + \Phi_{2,1}^{(1)}(\Phi_{1,1}^{(1)})^{-1}(r_1 - \theta_1^{(1)}) \\ &= 2.5 + 0.25 \times (1.6)^{-1}(r_1 - 3.2), \\ \theta_3^{(2)}(r_1) &= \theta_3^{(1)} + \Phi_{3,1}^{(1)}(\Phi_{1,1}^{(1)})^{-1}(r_1 - \theta_1^{(1)}) \\ &= 3.5 + 1.6 \times (1.6)^{-1}(r_1 - 3.2). \end{aligned}$$

Therefore, when $r_1 < 2.01$ we should choose user 2 in the second stage whilst when $r_1 > 2.01$ we should choose user 3 (when $r_1 = 2.01$ choosing either will give the same expected reward in the second stage). The corresponding value function is

$$\begin{aligned} V_{u=1}^*(\theta^{(1)}, \Phi^{(1)}, 2) &= \theta_1^{(1)} + \int p^{(1)}(r_1) \cdot \max_{j=2,3} \left(\theta_j^{(1)} + \Phi_{j,1}^{(1)}(\Phi_{1,1}^{(1)})^{-1}(r_1 - \theta_1^{(1)}) \right) dr_1 \\ &= 3.2 + \int_{-\infty}^{2.01} p^{(1)}(r_1) (2.5 + 0.25 \times (1.6)^{-1}(r_1 - 3.2)) dr_1 \\ &\quad + \int_{2.01}^{+\infty} p^{(1)}(r_1) (3.5 + 1.60 \times (1.6)^{-1}(r_1 - 3.2)) dr_1 \\ &\approx 6.80. \end{aligned}$$

Similarly, we can obtain the value functions for choosing user 2 and 3 at the initial stage

$$\begin{aligned} V_{u=2}^*(\theta^{(1)}, \Phi^{(1)}, 2) &\approx 5.7, \\ V_{u=3}^*(\theta^{(1)}, \Phi^{(1)}, 2) &\approx 6.77. \end{aligned}$$

And thus obtain the final value function

$$V^*(\theta^{(1)}, \Phi^{(1)}, 2) = \max(6.80, 5.7, 6.77) = 6.80.$$

We can see that the value function favours the first user at the first step, even though the prior information about the users favours the third user over the first user. Due to the fact that user 1 is highly correlated to user 3, and is more correlated with user 2 than user 3 is, choosing user 1 at the initial stage will enable the system to judge better in the second stage which results in a higher total expected reward over the two stages.

2.4 Computational Complexity

The exact solution of a finite-horizon POMDP has been proven to be PSPACE-complete [25]. In our case, the decision space at the initial stage is $C_m^{|\mathcal{U}|}$. For each decision, the m -dimensional observation space will be divided into $C_n^{|\mathcal{U}|-m}$ regions, each region corresponds to a (possibly) different optimal user combination to choose for the second stage. That is, the exact solution suggested by the value iteration algorithm requires going through all the possible decisions and all possible observations, which is intractable.

3 Approximation

To ease the intractability of the exact solution, we propose an approximation solution here, named guided exploitation-exploration (GEE). We provide its form for both the CU model and the MF model below.

3.1 Approximation for CU-POMDP

From Section 2.1.2, we have seen that the merit of selecting a group of users lies both in the immediate reward term (the exploitation part of Eq. (17)) and in how it can guide the system to find promising users in the next stage through the system update (the exploration part of Eq. (17)). However, when the decision of the initial stage is made, the system's belief state update is unknown before receiving any observations. To investigate the influence of selecting users \mathbf{u} only (before making any observations), let us consider the conditional distribution of unselected users $\setminus \mathbf{u}$ over the selection of users \mathbf{u} , $p(\mathbf{R}_{\setminus \mathbf{u}}|\mathbf{u})$. Note that this conditional distribution is different from Eq. (8) because it is the distribution conditioned on the action \mathbf{u} instead of the observations, as at the initial-decision stage these observations are still unknown.

Because the observations are not made yet, the expected feedback conditioned on the selection remains unchanged

$$\mathbb{E}[\mathbf{R}_{\setminus \mathbf{u}}|\mathbf{u}] = \boldsymbol{\theta}_{\setminus \mathbf{u}}^{(1)}. \quad (26)$$

However, its covariance changes according to the choice of \mathbf{u} :

$$\begin{aligned} \text{Cov}[\mathbf{R}_{\setminus \mathbf{u}}|\mathbf{u}] &= \text{Cov} \left[\boldsymbol{\theta}_{\mathbf{u}}^{(1)} + \boldsymbol{\Phi}_{\setminus \mathbf{u}, \mathbf{u}}^{(1)} (\boldsymbol{\Phi}_{\mathbf{u}, \mathbf{u}}^{(1)})^{-1} (\mathbf{R}_{\mathbf{u}} - \boldsymbol{\theta}_{\mathbf{u}}^{(1)}) \right] \\ &= \boldsymbol{\Phi}_{\setminus \mathbf{u}, \mathbf{u}}^{(1)} (\boldsymbol{\Phi}_{\mathbf{u}, \mathbf{u}}^{(1)})^{-1} \text{Cov}(\mathbf{R}_{\mathbf{u}}) (\boldsymbol{\Phi}_{\mathbf{u}, \mathbf{u}}^{(1)})^{-1} \boldsymbol{\Phi}_{\mathbf{u}, \setminus \mathbf{u}}^{(1)} \\ &= \boldsymbol{\Phi}_{\setminus \mathbf{u}, \mathbf{u}}^{(1)} (\boldsymbol{\Phi}_{\mathbf{u}, \mathbf{u}}^{(1)})^{-1} \boldsymbol{\Phi}_{\mathbf{u}, \setminus \mathbf{u}}^{(1)}, \end{aligned} \quad (27)$$

where the last step is due to $\text{Cov}(\mathbf{R}_{\mathbf{u}}) = \boldsymbol{\Phi}_{\mathbf{u}, \mathbf{u}}^{(1)}$.

Therefore, with the initial-stage users as \mathbf{u} , the expected returns at the second stage by choosing users \mathbf{v} are bounded by the interval $\Theta_{\mathbf{u}, \mathbf{v}}$:

$$\begin{aligned} \Theta_{\mathbf{u}, \mathbf{v}} &= \left[\mathbf{1}^T \left(\boldsymbol{\theta}_{\mathbf{v}}^{(1)} - \lambda \cdot \text{diag} \left[\text{Dg}^{-\frac{1}{2}} (\text{Cov}(\mathbf{R}_{\mathbf{v}}|\mathbf{u})) \right] \right), \right. \\ &\quad \left. \mathbf{1}^T \left(\boldsymbol{\theta}_{\mathbf{v}}^{(1)} + \lambda \cdot \text{diag} \left[\text{Dg}^{-\frac{1}{2}} (\text{Cov}(\mathbf{R}_{\mathbf{v}}|\mathbf{u})) \right] \right) \right] \end{aligned} \quad (28)$$

Algorithm 1 CU-GEE by Sampling

Require: Prior mean ratings $\boldsymbol{\theta}^{(1)}$, covariance matrix $\boldsymbol{\Phi}^{(1)}$, GEE parameter λ , available users \mathcal{U}

Initialise $\mathbf{u}^* \leftarrow \emptyset$

for $t = 1 \dots T$ **do**

 Sample \mathbf{u}_t ($|\mathbf{u}_t| = m$) from \mathcal{U}

 Calculate $V_{\mathbf{u}_t}^{\text{CU-GEE}}$ according to Eq. (29)

if $V_{\mathbf{u}_t}^{\text{CU-GEE}}$ is the largest so far **then**

 Update $\mathbf{u}^* \leftarrow \mathbf{u}_t$

end if

end for

with the probability at least $(1 - 2e^{-\lambda^2/2})^n$ [42]¹.

The GEE algorithm therefore optimistically assumes the highest return could be achieved within this interval [43]. And thus we choose the users \mathbf{u} which can achieve the highest total ratings under this assumption

$$\begin{aligned} \pi_{\text{CU-GEE}}(\boldsymbol{\theta}^{(1)}, \boldsymbol{\Phi}^{(1)}, \mathcal{U}) \\ = \arg \max_{\mathbf{u} \subset \mathcal{U}} \left\{ \mathbf{1}^T \boldsymbol{\theta}_{\mathbf{u}}^{(1)} + \max_{\mathbf{v} \subset \mathcal{U} \setminus \mathbf{u}} \mathbf{1}^T \left(\boldsymbol{\theta}_{\mathbf{v}}^{(1)} + \right. \right. \\ \left. \left. \lambda \cdot \text{diag} \left[\text{Dg}^{-\frac{1}{2}} \left(\boldsymbol{\Phi}_{\mathbf{v}, \mathbf{u}}^{(1)} (\boldsymbol{\Phi}_{\mathbf{u}, \mathbf{u}}^{(1)})^{-1} \boldsymbol{\Phi}_{\mathbf{u}, \mathbf{v}}^{(1)} \right) \right] \right) \right\}. \end{aligned} \quad (29)$$

This algorithm suggests that, in order to determine the users for stage one, we first calculate the immediate reward based on the prior information. Then we calculate the optimistic reward when acting optimally in the second stage. We call GEE *guided* as the initial-stage decision is optimistically guided by pseudo optimal user selections in the next stage. By inspecting into the next stage, we utilise the correlation between users of the two stages, which will be explained further in Section 3.1.1. To implement this algorithm, we can adopt a sampling-based method depicted in Algorithm 1.

3.1.1 Independent Intra-Stage User Assumption To align our algorithm with the popular memory-based CF, we adopt the correlation function Eq. (2) and reformulate Eq. (29) as follows:

$$\begin{aligned} & \boldsymbol{\Phi}_{\mathbf{v}, \mathbf{u}}^{(1)} (\boldsymbol{\Phi}_{\mathbf{u}, \mathbf{u}}^{(1)})^{-1} \boldsymbol{\Phi}_{\mathbf{u}, \mathbf{v}}^{(1)} \\ = & [\text{diag}(\boldsymbol{\phi}_{\mathbf{v}}^{(1)}) \mathbf{C}_{\mathbf{v}, \mathbf{u}}^{(1)} \text{diag}(\boldsymbol{\phi}_{\mathbf{u}}^{(1)})] [\text{diag}^{-1}(\boldsymbol{\phi}_{\mathbf{u}}^{(1)}) (\mathbf{C}_{\mathbf{u}, \mathbf{u}}^{(1)})^{-1} \text{diag}^{-1}(\boldsymbol{\phi}_{\mathbf{u}}^{(1)})] [\text{diag}(\boldsymbol{\phi}_{\mathbf{u}}^{(1)}) \mathbf{C}_{\mathbf{u}, \mathbf{v}}^{(1)} \text{diag}(\boldsymbol{\phi}_{\mathbf{v}}^{(1)})] \\ = & \text{diag}(\boldsymbol{\phi}_{\mathbf{v}}^{(1)}) \mathbf{C}_{\mathbf{v}, \mathbf{u}}^{(1)} (\mathbf{C}_{\mathbf{u}, \mathbf{u}}^{(1)})^{-1} \mathbf{C}_{\mathbf{u}, \mathbf{v}}^{(1)} \text{diag}(\boldsymbol{\phi}_{\mathbf{v}}^{(1)}). \end{aligned} \quad (30)$$

Eq. (29) thus becomes

$$\begin{aligned} \pi_{\text{CU-GEE}'}(\boldsymbol{\theta}^{(1)}, \boldsymbol{\phi}^{(1)}, \mathbf{C}^{(1)}, \mathcal{U}) \\ = \arg \max_{\mathbf{u} \subset \mathcal{U}} \left\{ \mathbf{1}^T \boldsymbol{\theta}_{\mathbf{u}}^{(1)} + \max_{\mathbf{v} \subset \mathcal{U} \setminus \mathbf{u}} \mathbf{1}^T \left(\boldsymbol{\theta}_{\mathbf{v}}^{(1)} + \right. \right. \\ \left. \left. \lambda \cdot \text{diag} \left[\text{Dg}^{-\frac{1}{2}} \left(\text{diag}(\boldsymbol{\phi}_{\mathbf{v}}^{(1)}) \mathbf{C}_{\mathbf{v}, \mathbf{u}}^{(1)} (\mathbf{C}_{\mathbf{u}, \mathbf{u}}^{(1)})^{-1} \mathbf{C}_{\mathbf{u}, \mathbf{v}}^{(1)} \text{diag}(\boldsymbol{\phi}_{\mathbf{v}}^{(1)}) \right) \right] \right) \right\}. \end{aligned} \quad (31)$$

The term of $(\mathbf{C}_{\mathbf{u}, \mathbf{u}}^{(1)})^{-1}$ in the above equation suggests us to diversify the items in the initial stage. Here in order to catch the more important relation between the

¹To be more exact, the conditional vector $\mathbf{R}_{\setminus \mathbf{u}} | \mathbf{u}$ is bounded in an ellipsoid. This form is obtained with an approximation of considering only the diagonal elements of $\text{Cov}(\mathbf{R}_{\setminus \mathbf{u}} | \mathbf{u})$.

Algorithm 2 CU-GEE-I by Sampling

Require: Prior mean ratings $\boldsymbol{\theta}^{(1)}$, correlation matrix $\mathbf{C}^{(1)}$, GEE parameter λ' , available users \mathcal{U}

Initialise $\mathbf{u}^* \leftarrow \emptyset$

for $t = 1 \dots T$ **do**

 Sample \mathbf{u}_t ($|\mathbf{u}_t| = m$) from \mathcal{U}

 Calculate $V_{\mathbf{u}_t}^{\text{CU-GEE-I}}$ according to Eq. (32)

if $V_{\mathbf{u}_t}^{\text{CU-GEE-I}}$ is the largest so far **then**

 Update $\mathbf{u}^* \leftarrow \mathbf{u}_t$

end if

end for

two stages, we assume the initial-stage users \mathbf{u} are independent of each other, which suggests an already-diversified user list. In addition to the independent assumption, we also impose an equal variance assumption, i.e., all the users have the same variance ϕ'^2 (so $\text{diag}(\boldsymbol{\phi}_v^{(1)}) = \phi' \mathbf{I}$). With the two assumptions, Eq. (31) can be further approximated to

$$\begin{aligned} & \pi_{\text{CU-GEE-I}}(\boldsymbol{\theta}^{(1)}, \mathbf{C}^{(1)}, \mathcal{U}) \\ &= \arg \max_{\mathbf{u} \subset \mathcal{U}} \left[\sum_{\alpha=1}^m \theta_{u_\alpha}^{(1)} + \max_{\mathbf{v} \subset \mathcal{U} \setminus \mathbf{u}} \sum_{\beta=1}^n \left(\theta_{v_\beta}^{(1)} + \lambda' \sqrt{\sum_{\alpha=1}^m (\rho_{u_\alpha, v_\beta}^{(1)})^2} \right) \right], \end{aligned} \quad (32)$$

where $\lambda' = \lambda \phi'$, and $\rho_{u_\alpha, v_\beta}^{(1)}$ is just the correlation between u_α and v_β according to the prior information. The effect of inter-stage user-user correlations is shown clearly in the above formula. According to Eq. (32), given the user selection at the initial stage \mathbf{u} , we can foresee the optimistic return in the next stage through highly expected values (via $\theta_{v_\beta}^{(1)}$) and also highly correlated users (via the term $\sqrt{\sum_{\alpha=1}^m (\rho_{u_\alpha, v_\beta}^{(1)})^2}$). Identifying these users then guides the system to determine the user selection \mathbf{u}^* .

The sampling method for this algorithm is illustrated in Algorithm 2.

3.2 Approximation for MF-POMDP

With the MF model, the conditional covariance matrix of $\mathbf{R}_{\setminus \mathbf{u}}$ given the user selection \mathbf{u} is written as

$$\text{Cov}(\mathbf{R}_{\setminus \mathbf{u}} | \mathbf{u}) = \mathbf{P}_{\setminus \mathbf{u}} \boldsymbol{\Psi}^{(1)} \mathbf{P}_{\mathbf{u}}^T (\mathbf{P}_{\mathbf{u}} \boldsymbol{\Psi}^{(1)} \mathbf{P}_{\mathbf{u}}^T + \sigma_0^2 \mathbf{I})^{-1} \mathbf{P}_{\mathbf{u}} \boldsymbol{\Psi}^{(1)} \mathbf{P}_{\setminus \mathbf{u}}^T. \quad (33)$$

Following the same reasoning as in Section 3.1, we give the formulation for the matrix factorization model

$$\begin{aligned} & \pi_{\text{MF-GEE}}(\boldsymbol{\nu}^{(1)}, \boldsymbol{\Psi}^{(1)}, \mathcal{U}) \\ &= \arg \max_{\mathbf{u} \subset \mathcal{U}} \left\{ \mathbf{1}^T \mathbf{P}_{\mathbf{u}} \boldsymbol{\nu}^{(1)} + \max_{\mathbf{v} \subset \mathcal{U} \setminus \mathbf{u}} \mathbf{1}^T \left(\mathbf{P}_{\mathbf{v}} \boldsymbol{\nu}^{(1)} + \lambda \cdot \right. \right. \\ & \quad \left. \left. \text{diag} \left[\text{Dg}^{-\frac{1}{2}} \left(\mathbf{P}_{\mathbf{v}} \boldsymbol{\Psi}^{(1)} \mathbf{P}_{\mathbf{u}}^T (\mathbf{P}_{\mathbf{u}} \boldsymbol{\Psi}^{(1)} \mathbf{P}_{\mathbf{u}}^T + \sigma_0^2 \mathbf{I})^{-1} \mathbf{P}_{\mathbf{u}} \boldsymbol{\Psi}^{(1)} \mathbf{P}_{\mathbf{v}}^T \right) \right] \right) \right\} \end{aligned} \quad (34)$$

The corresponding algorithm is shown in Algorithm 3.

Algorithm 3 MF-GEE by Sampling

Require: Prior mean $\boldsymbol{\nu}^{(1)}$ and covariance matrix $\boldsymbol{\Psi}^{(1)}$ of the target item feature vector, GEE parameter λ , available users \mathcal{U}
Initialise $\mathbf{u}^* \leftarrow \emptyset$
for $t = 1 \dots T$ **do**
 Sample \mathbf{u}_t ($|\mathbf{u}_t| = m$) from \mathcal{U}
 Calculate $V_{\mathbf{u}_t}^{\text{MF-GEE}}$ according to Eq. (34)
 if $V_{\mathbf{u}_t}^{\text{MF-GEE}}$ is the largest so far **then**
 Update $\mathbf{u}^* \leftarrow \mathbf{u}_t$
 end if
end for

3.2.1 Independent Intra-Stage User Assumption With the MF model, in addition to the independent intra-stage user assumption which turns $\mathbf{P}_u \boldsymbol{\Psi}^{(1)} \mathbf{P}_u^T$ into a diagonal matrix, we may also assume independent latent dimensions such that the prior covariance matrix is diagonal: $\boldsymbol{\Psi}^{(1)} = \text{diag}^2[\boldsymbol{\psi}^{(1)}]$, where $\boldsymbol{\psi}^{(1)}$ are the standard deviations of latent dimensions. Eq. (34) can be further simplified as:

$$\pi_{\text{MF-GEE-I}}(\boldsymbol{\nu}^{(1)}, \boldsymbol{\psi}^{(1)}, \mathcal{U}) = \arg \max_{\mathbf{u} \subset \mathcal{U}} \left\{ \sum_{\alpha=1}^m \mathbf{p}_{u_\alpha}^T \boldsymbol{\nu}^{(1)} + \max_{\mathbf{v} \subset \mathcal{U} \setminus \mathbf{u}} \sum_{\beta=1}^n \left(\mathbf{p}_{v_\beta}^T \boldsymbol{\nu}^{(1)} + \lambda \sqrt{\sum_{\alpha=1}^m \frac{(\mathbf{p}_{v_\beta}^T \text{diag}^2[\boldsymbol{\psi}^{(1)}] \mathbf{p}_{u_\alpha})^2}{\mathbf{p}_{u_\alpha}^T \text{diag}^2[\boldsymbol{\psi}^{(1)}] \mathbf{p}_{u_\alpha} + \sigma_0^2}} \right) \right\}. \quad (35)$$

The corresponding algorithm is shown in Algorithm 4.

Particularly, when assuming $\boldsymbol{\psi}^{(1)} = \psi^{(1)} \mathbf{1}$, i.e., equal prior standard deviation (variance) along different dimensions, we gain the form

$$\pi_{\text{MF-GEE-II}}(\boldsymbol{\nu}^{(1)}, \boldsymbol{\psi}^{(1)}, \mathcal{U}) = \arg \max_{\mathbf{u} \subset \mathcal{U}} \left\{ \sum_{\alpha=1}^m \mathbf{p}_{u_\alpha}^T \boldsymbol{\nu}^{(1)} + \max_{\mathbf{v} \subset \mathcal{U} \setminus \mathbf{u}} \sum_{\beta=1}^n \left(\mathbf{p}_{v_\beta}^T \boldsymbol{\nu}^{(1)} + \lambda \sqrt{\sum_{\alpha=1}^m \frac{((\psi^{(1)})^2 \mathbf{p}_{v_\beta}^T \mathbf{p}_{u_\alpha})^2}{(\psi^{(1)})^2 \mathbf{p}_{u_\alpha}^T \mathbf{p}_{u_\alpha} + \sigma_0^2}} \right) \right\}. \quad (36)$$

Actually, with such a spherical prior variance, Eq. (24) becomes $\Phi_{u,v}^{(1)} = (\psi^{(1)})^2 \mathbf{p}_u^T \mathbf{p}_v$, i.e., the covariance between u and v is proportional to the inner product of the user latent factors. Actually, with a spherical prior variance, the correlation between user u and v , $\rho_{u,v}$, is proportional to $\mathbf{p}_u^T \mathbf{p}_v$, corresponding to the MF obtained by a regularised linear regression estimation [30].

4 Related Work and Discussion

4.1 Collaborative Filtering

Our work can be considered part of CF research [39]. CF provides efficient and personalised recommendations based on the similarities between users and items. This can be achieved by mainly three approaches [39]: a similarity-based approaches such as neighbourhood based CF (user-based and item-based) [17,10], latent factor models [22,21,21,7], and hybrid methods [9]. We relate our work with the neighbourhood-based CF and latent factor models as follows.

Algorithm 4 MF-GEE-I by Sampling

Require: Prior mean $\nu^{(1)}$ and the diagonal element of the covariance matrix $\psi^{(1)}$ of the target item feature vector, GEE parameter λ , available users \mathcal{U}
Initialise $\mathbf{u}^* \leftarrow \emptyset$
for $t = 1 \dots T$ **do**
 Sample \mathbf{u}_t ($|\mathbf{u}_t| = m$) from \mathcal{U}
 Calculate $V_{\mathbf{u}_t}^{\text{MF-GEE-I}}$ according to Eq. (35)
 if $V_{\mathbf{u}_t}^{\text{MF-GEE-I}}$ is the largest so far **then**
 Update $\mathbf{u}^* \leftarrow \mathbf{u}_t$
 end if
end for

4.1.1 Neighbourhood-Based CF Neighbourhood based CF provides a straightforward estimation of the target rating as a weighted summation of similar ratings: either the ratings from similar users, or the ratings to similar items, and can therefore provide explainable recommendations [17,16,1,34,6,30].

According to our analysis in Section 2.1, neighbourhood-based models can be viewed as an approximate multivariate Gaussian preference model with the following two assumptions: (i) only the correlations between the target user and its neighbours are considered, and the neighbour users are assumed to be independent to each other; and (ii) all users have the same variance in their rating behaviours. In some practices, the rating scores are also normalised with their standard deviations [16], which is referred to as the Z-score normalisation. We thus see the Z-score normalisation as a way to alleviate the prediction discrepancy caused by the second assumption. In addition, in practice, only the most-similar users are selected as neighbours, including top- N filtering and threshold filtering strategies, to ease the computational cost [30]. Beside the Pearson correlation, Cosine vector similarity is also used, but it is argued that its performances are not as good as the Pearson correlation similarity measure [8].

4.1.2 Latent Factor Models Latent factor models first project the user and item onto a latent feature space, and then base the score on the feature vectors of them. The correlations between user pairs are therefore translated as the vector similarity in the latent space (as shown in Section 2.2). Matrix factorisation is probably the most well-known method of latent factor models [22]. Singular value decomposition (SVD) [22], SVD++ [21], pLSA [18] and Latent Dirichlet Allocation [7] are among the more famous ones. Probabilistic matrix factorisation (MF) is one of the latent factors which is adopted in this paper [33] with respect its probabilistic property.

4.2 Cold-Start Problems in CF

Cold-start problems [35] remain a major challenge for CF-based recommender systems, as the prediction of ratings purely depends on the previously expressed user-item preferences without the use of any content information, and for a new user or item this information is unavailable.

There is comparatively more literature on the user cold-start problems than the item cold-start problems. For the former, an pre-recommendation ‘interview’ process is usually adopted. In an interview, the user first gives feedback on some questions provided by the system, such as preferences on some popular items or highly informative items, or on a diversified list for the users to rate [26,27]. The interview phase can

also be more intelligent, as decision-tree based methods suggest [26,48,12]. AL forms an important branch for designing interview questions, and is most relevant to our approach. We have a thorough comparison in Section 4.4.1.

In addition, many techniques [48,12,47] assume an interactive process for sequential query selection, i.e., only one query is chosen at a time for one user. Then after the response is collected, another query will be chosen according to the user response to the previous query. In our case, as well as in many other practical situations, multiple items or users should be recommended in a batch manner to improve efficiency. Therefore, iterative techniques are not applicable.

4.3 Probabilistic Ranking Principle in CF

The well-known probabilistic ranking principle (PRP) has been related to CF [45], which suggests that the top- N recommendation list can be generated by ranking according to the probability of relevance to the target (a user or an item). Originated from information retrieval [29], PRP implies documents to be ranked in descending order by their probability of relevance can produce optimal performance under the “independent document” assumption [28]. In the item cold-start problem scenario, supposing the rating is proportional to the relevance probability, the list of users to recommend the item to should be ranked according to the prior information of the users, such as the rank of the user average ratings. On the other hand, in a user cold-start scenario, the rank of recommended items should be the prior average rating information of the items.

We have shown in this paper that PRP is not optimal as the correlations between users play an important role for the system to update, when considered as an interactive process. There are both intra-list correlations between users chosen in the initial stage and inter-list correlations between users in the first and remaining users (Eq. 17). Especially, the inter-list correlations enable the system to update, and finally lead to more accurate predictions.

4.4 Comparisons to Other EE Methods

4.4.1 Comparison to Active Learning Active learning (AL) methods have been adopted to handle cold-start problems in recommender systems [30,15,32,31], which are also referred to as optimal design by statisticians [40]. AL uses a limited number of items (usually much smaller than the total number of available items) to present to the target user to review, and then learns the user’s profile based on the users’ feedback on these items. The criterion for selection is usually represented by a statistical measure such as achieving minimal mean squared error in the model estimation (A-optimality criterion) [2], minimal 2-norm of the inverse of the information matrix (E-optimality criterion) [32] or minimal determinant of resulting covariance matrix of the system (D-optimality criterion) [32]. This objective differs from our objective function, and thus leads to significant differences from our approach.

There are two main differences between AL and our GEE approach. First, AL techniques such as D-Optimal design [32], A-Optimal design [2] and their applications to the cold-start item problem have divided exploration and exploitation into two separate stages. In the exploration stage, a small number of training points are selected for the system to learn, and in the exploitation stage the gained information is fully exploited. However, the returns (or regrets) collected from the exploration stage are

not considered. In other words, The objective is imposed onto only the exploitation stage, and thus the trade-off between exploration and exploitation is not modeled [30]. For example, in [2], a budget has been imposed on the number of users to select at the experimental stage, and these users’ returns are excluded from the objective function.

Second, the goal of AL is usually measured statistically using a global criterion. The criterion can be, for example, (to minimise) the mean square error of the estimates [2], or, (to maximise) the differential Shannon information [32]. However, from Eqs. (17) and (25) and from the example, we can see that the exact solution is achieved by prioritising the learning process towards the promising users of the next stage. Therefore, it is not necessary to achieve a global optimum. On the contrary, GEE captures this feature and make decisions guided by potential users of the second stage.

4.4.2 Comparison to UCB methods The EE problem has been intensively studied in the literature of multi-armed bandit problems, where an agent decides dynamically which arm to choose at each step bearing the objective to maximise the total reward collected during a period of time [3]. Gittins has provided an optimal solution under the condition that only one arm at a time can evolve [11], but this is intractable in practice. UCB seeks a bounded regret instead of optimality and is used to balance the exploitation and exploration in practice [3,4,38,43]. In UCB, usually a decision is made based on both the expectation and uncertainty of the return of individual choices at each step. In [47], we proposed several UCB-based algorithms for a multiple-stage interactive recommendation process. And recently GP-UCB algorithms have also been applied to solve the user cold-start problems interactively in recommender systems [41].

Our approach differs from UCB approaches in the following ways. First, UCB-based approaches seek to limit the regret within a bound, but they do not model how the specific selection within the bound can influence the outcome. In other words, EE achieved by UCB is not guided by the potential rewarding choice of the following stage, but is rather to limit the regret of the current stage. Second, UCB-based approaches are usually achieved in a long-term and interactive process, and may not be suitable for the two-stage process. Conversely, our algorithms are derived directly from the exact solutions of POMDP. They have directly considered the effect that choosing the initial-stage users has on the potential returns from the second stage.

5 Experiment

In this section, we compare our proposed approximate solutions with several baseline methods. To understand the model further and verify our theoretical analysis, we first present the results on synthetic data, and then on a real dataset.

5.1 Synthetic Data Experiment

5.1.1 Synthetic Data Generation First, we define a 5-dimensional latent space and randomly generate a multivariate Gaussian distribution as the prior information of the cold-start item. In detail, each dimension of the multivariate Gaussian mean vector is generated randomly according to $\mathcal{N}(0, 0.1)$, and each dimension’s standard deviation is generated according to $\mathcal{N}(0, 1)$. Then we generate 50 cold-start items according to this randomly-generated distribution. Second, we generate 100

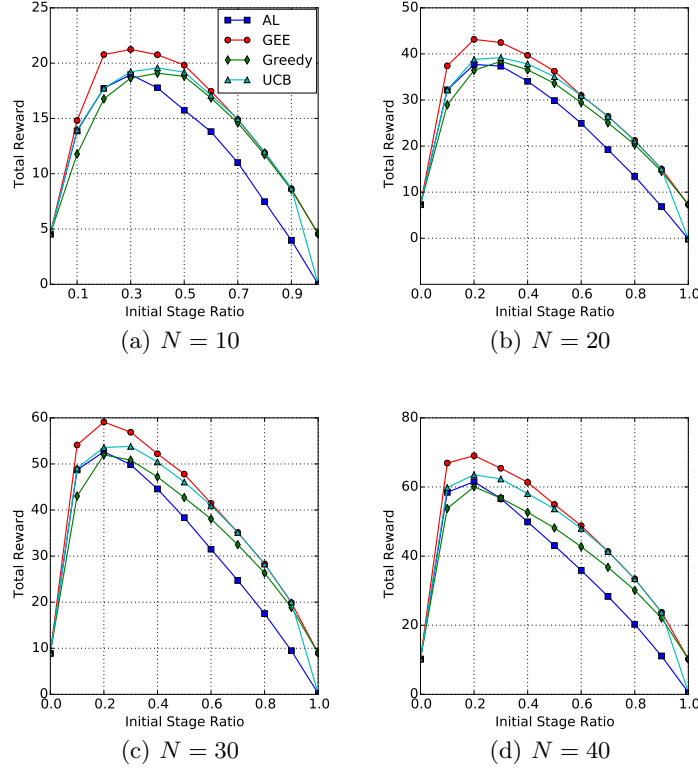


Fig. 4: Total reward comparison of different algorithms on the synthetic data. The x-axis is m/N , the ratio of users to choose at the initial stage, and the y-axis is the total reward of both stages.

users' vectors according to $\mathcal{N}(\mathbf{0}, \mathbf{I})$ as the available user pool for the 50 cold-start items to target to. Their real ratings are then produced according to Eq. (19) with the noise's standard deviation as 0.5. As such, we can obtain a 100×50 rating matrix as the groundtruth. The true prior information is then provided for each compared algorithm to perform recommendations. Finally, the above process is repeated for a total of 30 times, each time with a different prior information of the cold-start items. The results are then averaged over the different trials.

5.1.2 Compared Methods We compare our proposed GEE algorithm to the following algorithms. (i) **Greedy**. Greedy method chooses the initial-stage users with the highest expected feedback. (ii) **Active learning (AL)**. AL method chooses the users

Table 2: Total reward compared using synthetic data.

Algorithm	$N = 10$	$N = 20$	$N = 30$	$N = 40$
Greedy	19.084	38.919	52.517	60.55
AL	18.953	37.719	52.655	62.537
UCB	19.568	39.903	54.632	63.959
GEE	21.238	43.151	59.315	69.198
Improvement	8.5%	8.1%	8.6%	8.2%

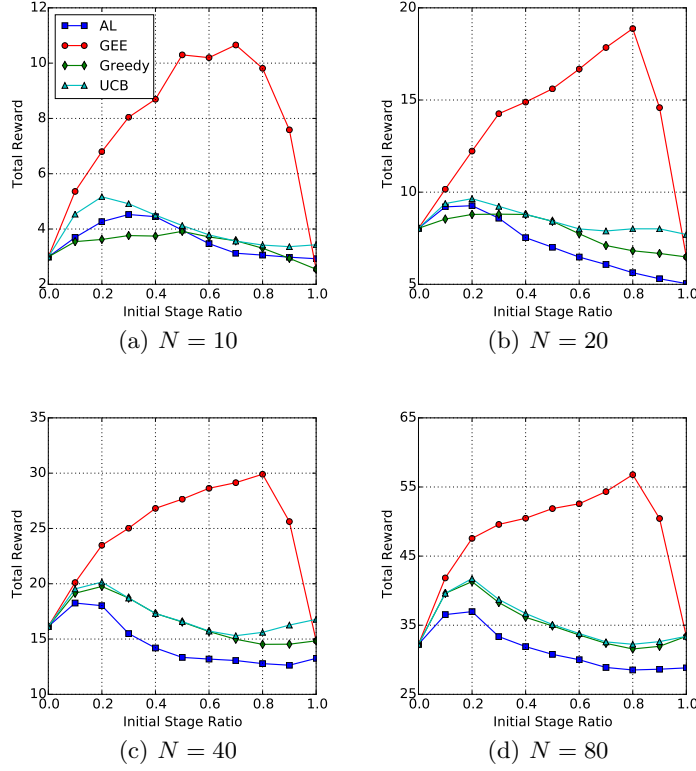


Fig. 5: Total reward comparison of different algorithms on the MovieLens 100K data. The x-axis is m/N , the ratio of users to choose at the initial stage, and the y-axis is the total reward of both stages.

to minimise the uncertainty in the model, so that the users with the highest variances are chosen [15,30]. (iii) Upper confidence bound (UCB). UCB method chooses the initial-stage users with the highest values calculated as the linear combination of the expected reward and the standard deviation [38]. All the algorithms select the second-stage users greedily after the system’s state is updated with observations.

5.1.3 Results The results are shown in Figure 4, with the evaluation measure as the total reward gained from the two stages. The result of the original GEE algorithm (Eq. (34)) is shown and we emphasise that the result of the GEE algorithm with the intra-stage independence assumption produces similar results.

From this figure, we can make the following observations. (i) For all the algorithms, the performance improves as m increases. This shows that by separating the recommendation process into two stages the performance can be greatly improved over a PRP-like once-for-all batch solution. (ii) For all the algorithms, the total reward increases more sharply than it drops after the performance peak. This phenomenon indicates that a small portion of allocation of users in the initial stage can significantly improve the overall performance. Note that in our synthetic data generation, we have used $K=5$, and the peak is also around $m = 5$. Therefore, the dimension of

Table 3: Total reward compared on MovieLens.

Algorithm	$N = 10$	$N = 20$	$N = 40$	$N = 80$
Greedy	4.255	8.95	20.75	45.26
AL	4.705	9.91	21.715	41.665
UCB	5.38	10.2	21.715	45.26
GEE	12.125	19.48	31.05	60.97
Improvement	125.4%	91.0%	43.0%	34.7%

Table 4: Total hit number compared on MovieLens.

Algorithm	$N = 10$	$N = 20$	$N = 40$	$N = 80$
Greedy	0.845	1.745	4.045	8.73
AL	0.875	1.905	4.155	7.815
UCB	1.015	1.955	4.155	8.73
GEE	2.245	3.245	5.325	10.225
Improvement	121.2%	66.0%	28.2%	17.1%

the latent factor model may be an indicator of the allocation ratio. The best result gained with optimal parameters of each algorithm is shown in Table 2.

5.2 Experiments on the MovieLens Dataset

5.2.1 Experiment setup As our study is a theoretical one, we use a relatively small research-based dataset MovieLens 100K, which is relatively small, containing 943 users and 1,682 movies, with altogether 100,000 ratings ranging from 1 to 5. To conduct the experiment, we first divide the dataset into the training set and test set. For the sake of simulating cold-start item recommendations, we first randomly choose 200 items with sufficient numbers of ratings (at least 50) as the test cold-start items, and use their ratings as the groundtruth in the test dataset. The ratings between users and the remaining items are used to train the model. Similar to the synthetic data experiment, we compare our algorithms with Greedy, AL and UCB. After observing the feedback, the system updates according to the user-based CF model suggested by Eq. (14). The results are evaluated by using both the total reward, and the total hit number – the total number of ratings equal or above 4 of the two stages. To be consistent with what the user-based CF model suggests, we use the independent intra-user assumption for the GEE algorithm used.

5.2.2 Results The results are shown in Figure 5, and Tables 3 and 4 with $N = 10, 20, 40$ and 80 respectively. Both the total reward and the total hit number measures are compared. Here the total hit number is defined as the total number of ratings collected which are 4 or above. We can see significant improvements over all four cases with the implementation of our algorithm. Similar to the synthetic experiment results, all algorithms show a peaking manner as m increases. From Tables 3 and 4 we can see that the improvements evaluated by using the total reward are even higher than the total hit number, which may be the result of targeting directly to the optimal reward in our objective function.

6 Conclusion and Future Work

In this paper, we presented a novel two-stage recommendation process to address the cold-start problems, with an item cold-start problem as a working example. We

formulated the problem using both a correlated-user model and a matrix factorisation model with POMDP. With the exact solution suggested by value iteration, we concluded that the users to choose at the initial stage should be not only of high expected values, but also highly correlated with potential users in the next stage – a property that can guide the system to find promising users in the next stage. With this finding, we proposed the approximate algorithm guided exploitation-exploration (GEE). We conducted initial experiments using GEE and compared the results with several baseline algorithms on both a synthetic and a real dataset, which confirmed the effectiveness of our algorithm.

For future work, we plan to extend the two-stage process to multiple stages and conduct larger scale experiments to study the scalability. We are also interested in obtaining the optimal trade-off parameter λ and the ratio of exploitation-exploration m/n theoretically.

References

1. G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *TKDE*, 2005.
2. O. Anava, S. Golan, N. Golbandi, Z. Karnin, R. Lempel, O. Rokhlenko, and O. Somekh. Budget-constrained item cold-start handling in collaborative filtering recommenders via optimal design. In *WWW*, 2015.
3. P. Auer. Using confidence bounds for exploitation-exploration trade-offs. *JMLR*, 2003.
4. P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 2002.
5. N. J. Belkin and W. B. Croft. Information filtering and information retrieval: two sides of the same coin? *Commun ACM*, 1992.
6. R. M. Bell and Y. Koren. Improved neighborhood-based collaborative filtering. In *SIGKDD Cup Workshop*, 2007.
7. D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *JMLR*, 2003.
8. J. S. Breese, D. Heckerman, and C. Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *UAI*, 1998.
9. R. Burke. Hybrid recommender systems: Survey and experiments. *UMUAI*, 2002.
10. M. Deshpande and G. Karypis. Item-based top-n recommendation algorithms. *TOIS*, 2004.
11. J. Gittins, K. Glazebrook, and R. Weber. *Multi-armed bandit allocation indices*. John Wiley & Sons, 2011.
12. N. Golbandi, Y. Koren, and R. Lempel. Adaptive bootstrapping of recommender systems using decision trees. In *WSDM*, 2011.
13. N. Good, J. B. Schafer, J. A. Konstan, A. Borchers, B. Sarwar, J. Herlocker, and J. Riedl. Combining collaborative filtering with personal agents for better recommendations. In *AAAI/IAAI*, 1999.
14. A. Gunawardana and C. Meek. Tied boltzmann machines for cold start recommendations. In *RecSys*, 2008.
15. A. S. Harpale and Y. Yang. Personalized active learning for collaborative filtering. In *SIGIR*, 2008.
16. J. Herlocker, J. A. Konstan, and J. Riedl. An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. *Information retrieval*, 2002.
17. J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl. An algorithmic framework for performing collaborative filtering. In *SIGIR*, 1999.

18. T. Hofmann. Latent semantic models for collaborative filtering. *TOIS*, 2004.
19. X. Jin, M. Sloan, and J. Wang. Interactive exploratory search for multi page search results. In *WWW*, 2013.
20. L. Kaelbling, M. Littman, and A. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 1998.
21. Y. Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *SIGKDD*, 2008.
22. Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 2009.
23. C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.
24. L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. 1999.
25. C. H. Papadimitriou and J. N. Tsitsiklis. The complexity of markov decision processes. *Mathematics of operations research*, 1987.
26. A. Rashid, I. Albert, D. Cosley, S. Lam, S. McNee, J. Konstan, and J. Riedl. Getting to know you: learning new user preferences in recommender systems. In *IUI*, 2002.
27. A. Rashid, G. Karypis, and J. Riedl. Learning preferences of new users in recommender systems: an information theoretic approach. *SIGKDD Explorations Newsletter*, 2008.
28. C. J. V. Rijsbergen. *Information Retrieval*. Butterworth-Heinemann, 2nd edition, 1979.
29. S. E. Robertson. The probability ranking principle in ir. *Journal of documentation*, 1977.
30. N. Rubens, D. Kaplan, and M. Sugiyama. Active learning in recommender systems. In *Recommender systems handbook*, pages 735–767. Springer, 2011.
31. N. Rubens and M. Sugiyama. Influence-based collaborative active learning. In *RecSys*, 2007.
32. N. Rubens, R. Tomioka, and M. Sugiyama. Output divergence criterion for active learning in collaborative settings. *IPSJ Online Transactions*, 2009.
33. R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. In *NIPS*, 2007.
34. B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In *WWW*, 2001.
35. A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock. Methods and metrics for cold-start recommendations. In *SIGIR*, 2002.
36. X. Shen, B. Tan, and C. Zhai. Implicit user modeling for personalized search. In *CIKM*, 2005.
37. Y. Shi, X. Zhao, J. Wang, M. Larson, and A. Hanjalic. Adaptive diversification of recommendation results via latent factor portfolio. In *SIGIR*, 2012.
38. N. Srinivas, A. Krause, S. M. Kakade, and M. Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*, 2009.
39. X. Su and T. M. Khoshgoftaar. A survey of collaborative filtering techniques. *Advances in artificial intelligence*, 2009:4, 2009.
40. G. Taguchi. *Introduction to quality engineering: designing quality into products and processes*. Asian Productivity Organization, 1986.
41. H. P. Vanchinathan, I. Nikolic, F. De Bona, and A. Krause. Explore-exploit in top-n recommender systems via gaussian processes. In *RecSys*, 2014.
42. F. G. Viens. Steins lemma, malliavin calculus, and tail bounds, with application to polymer fluctuation exponent. *Stoch Proc Appl*, 2009.

- 43. T. Walsh, I. Szita, C. Diuk, and M. Littman. Exploring compact reinforcement-learning representations with linear regression. In *UAI*, 2009.
- 44. J. Wang, A. P. De Vries, and M. J. Reinders. Unified relevance models for rating prediction in collaborative filtering. *TOIS*, 2008.
- 45. J. Wang, S. Robertson, A. P. de Vries, and M. J. Reinders. Probabilistic relevance ranking for collaborative filtering. *Information Retrieval*, 2008.
- 46. W. Zhang, S. Yuan, and J. Wang. Optimal real-time bidding for display advertising. In *SIGKDD*, 2014.
- 47. X. Zhao, W. Zhang, and J. Wang. Interactive collaborative filtering. In *CIKM*, 2013.
- 48. K. Zhou, S. Yang, and H. Zha. Functional matrix factorizations for cold-start recommendation. In *SIGIR*, 2011.